Decryptable Searchable Encryption

Thomas Fuhr[1] and Pascal Paillier[2]

[1] Direction Centrale de la Sécurité des Systèmes d'Information
thomas.fuhr@sgdn.pm.gouv.fr
[2] Cryptography & Innovation, Gemalto Security Labs
pascal.paillier@gemalto.com

Abstract. As such, public-key encryption with keyword search (a.k.a PEKS or searchable encryption) does not allow the recipient to decrypt keywords *i.e.* encryption is not invertible. This paper introduces searchable encryption schemes which enable decryption. An additional feature is that the decryption key and the trapdoor derivation key are *totally independent*, thereby complying with many contexts of application. We put forward a seemingly optimal construction for decryptable searchable encryption which makes use of one KEM, one IDKEM and a couple of hash functions. We define a proper security model for decryptable searchable encryption and show that basic security requirements on the underlying KEM and IDKEM are enough for our generic construction to be strongly secure in the random oracle model.

1 Introduction

Background. Among the most recent developments of public-key cryptography, the mechanisms for ID-based encryption [19,5,6,13,3] and public-key encryption with keyword search (PEKS) have become increasingly attractive thanks to their connections with many other (still unsolved) design issues. It seems that the idea of encryption with keyword search, also known as *searchable encryption* [4], appeared as a natural application of what one could achieve with bilinear maps, which already provided the basis for ID-based encryption. A more recent work [1] shows that these mechanisms are intimately related in the sense that they are induced by a common primitive known as an anonymous IDKEM [8].

Informally, a searchable encryption c of a keyword w can only be tested by the recipient who uses her private key to detect whether c matches w or not. This ability is transferrable to anyone under the form of a keyword-specific trapdoor $\mathsf{T}(w)$ which enables the search for encryptions of w. In a typical application of searchable encryption, the entity holding $\mathsf{T}(w)$ receives lots of encrypted keywords and filters out encryptions of $w' \neq w$. Searchable encryption, as currently achieved, does not require ciphertexts to be decryptable.

Our Contributions. This paper introduces searchable encryption schemes that enable decryption. We mention that the decryption key and the trapdoor derivation key are independent of each other, thereby complying with various contexts

W. Susilo, J.K. Liu, and Y. Mu. (Eds.): ProvSec 2007, LNCS 4784, pp. 228–236, 2007.

of application. We put forward a generic construction for decryptable searchable encryption. To achieve our goal, we make use of generic cryptographic primitives such as key encapsulation mechanisms (a.k.a. KEMs) [10] and identity-based versions of KEMs (IDKEMs). Our construction also employs a couple of hash functions. We define a proper security model for decryptable searchable encryption and investigate under which security requirements on the underlying KEM and IDKEM blocks our construction yields a maximally secure scheme. All security proofs considered in this paper stand in the random oracle model.

Applications of Our Work. Decryptable searchable encryption (DSE for short) extends the notion of PEKS and may therefore be used in every single application of PEKS. We may also find applications in the management of encrypted databases. In particular, since the decryption key and the trapdoor derivation key are generated independently from one another, data can be decrypted by an entity and trapdoors be generated by some other party. An illustrative example of this feature is as follows. Assume Bob is a telephone operator, Alice a subscriber, Charlie a state agency and Daniel a police inspector whose role consists in identifying subscribers belonging to the Mafia. Assume that Bob stores Alice's telephone statement encrypted with DSE, and that the decryption key belongs to Alice and the trapdoor derivation key belongs to Charlie. Alice is the only person who can decrypt it, but Charlie can issue trapdoors for some phone numbers and give them to Daniel to help him find out whether Alice is connected to the Mafia, without learning anything about the other numbers Alice has called. The same scenario is applicable to the secure management of money transfers, wherein a maximal level of secrecy about account numbers involved in transactions is guaranteed, while leaving to a designated authority the ability to trace encrypted transactions made to or from well-identified bank accounts.

Outline. We start in Section 2 by a number of definitional facts about KEMs and IDKEMs. Section 3 describes our generic construction and provides a security analysis. Section 4 provides an example of instantiation based on ElGamal and BDOP [4]. Section 5 concludes on a number of questions left open by this work.

2 Preliminaries on Encapsulation Mechanisms

2.1 Key Encapsulation Mechanisms (KEMs)

Definition. A KEM is a basic cryptographic primitive by the means of which one can publicly and securely encapsulate a randomly generated session key. The owner of the private key (the decapsulation key) can later recover the session key given the encapsulation. KEMs make use of decapsulation keys, encapsulation keys, random numbers, ciphertexts and secret values (that may be symmetric keys). Here we will not describe their inner structure, but rather give a general description of the primitive. We identify a KEM to a tuple of probabilistic algorithms $\mathsf{KEM} = (\mathsf{KEM.Gen}, \mathsf{KEM.Encap}, \mathsf{KEM.Decap})$ defined as follows.